

## CLAIMS

WHAT IS CLAIMED IS:

1. A method of creating programmable data objects for use in a multi-tier computing architecture, the method comprising:
- dragging a graphical representation for a server processing resource from a server explorer module to a visual design surface module to add a processing item to a programmable data object being created in the visual design surface module;
  - identifying data schema associated with the server processing resource added to the programmable data object;
  - creating a typed dataset containing the data structures corresponding to the data schema associated with the server processing resource;
  - creating a command adapter to provide data transfer commands within the programmable data object between the programmable data object and the server processing resource; and
  - creating a data transfer connection between the programmable data object and the server processing resource.
2. The method according to claim 1, wherein the method further comprises: creating additional initialization methods to support the items added to the programmable data object.
3. The method according to claim 2, wherein the method further comprises: editing the processing items within the visual design surface module; updating the processing items edited within visual design surface module;

*Handwritten signature/initials*

- updating the typed dataset within visual design surface module;
  - updating the command adapters within the visual design surface module;
  - identifying any other processing items containing references to data structures and functions edited; and
  - updating the identified items containing references to data structures and functions edited to make all references consistent with each other.
4. The method according to claim 3, wherein the items comprise properties and processing instruction source code that may be edited.
  5. The method according to claim 1, wherein the method further comprises:
    - inserting a database connection module that creates a data transfer connection between the programmable data object and a database when the dragged item is database table within the database.
  6. The method according to claim 5, wherein the database connection module comprises:
    - a data connection object for creating and managing the data transfer connection between the programmable data object and the database;
    - a managed resource module for provides the data connection object with address and identification information to establish the data transfer connection; and
    - a persistent data storage for maintaining this address and identification information used by the managed resource module.
  7. A computing system of creating programmable data objects for use in a multi-tier computing architecture, the computing system comprising:

Sub  
ar

a memory module;  
a user interface module;  
a mass storage system; and  
a programmable processing module, the programmable processing module performing a sequence of operations to implement the following:

dragging a graphical representation for a server processing resource from a server explorer module to a visual design surface module to add a processing to a programmable data object being created in the visual design surface module;

identifying data schema associated with the server processing resource added to the programmable data object;

creating a typed dataset containing the data structures corresponding to the data schema associated with the server processing resource;

creating a command adapter to provide data transfer commands within the programmable data object between the programmable data object and the server processing resource; and

creating a data transfer connection between the programmable data object and the server processing resource.

8. The computing system according to claim 7, wherein the sequence of operations further comprises creating additional initialization methods to support the items added to the programmable data object.

9. A computing system according to claim 8, wherein the sequence of operations further comprises:

Sub  
ar

editing the processing items within the visual design surface module;  
updating the processing items edited within visual design surface module;  
updating the typed dataset within visual design surface module;  
updating the command adapters within the visual design surface module;  
identifying any other processing items containing references to data structures  
and functions edited; and

updating the identified items containing references to data structures and  
functions edited to make all references consistent with each other.

10. The computing system according to claim 9, wherein the items comprise  
properties and processing instruction source code that may be edited.

11. The computing system according to claim 10, wherein the sequence of  
operations further comprises:

inserting a database connection module that creates a data transfer connection  
between the programmable data object and a database when the dragged item is  
database table within the database.

12. The computing system according to claim 11, wherein the database connection  
module comprises:

a data connection object for creating and managing the data transfer connection  
between the programmable data object and the database;

a managed resource module for provides the data connection object with address  
and identification information to establish the data transfer connection; and

a persistent data storage for maintaining this address and identification

information used by the managed resource module.

13. A computer program product readable by a computing system and encoding instructions for a computing process for creating programmable data objects for use in a multi-tier computing architecture the computing process comprising:

dragging a graphical representation for a server processing resource from a server explorer module to a visual design surface module to add a processing item to a programmable data object being created in the visual design surface module;

identifying data schema associated with the server processing resource added to the programmable data object;

creating a typed dataset containing the data structures corresponding to the data schema associated with the server processing resource;

creating a command adapter to provide data transfer commands within the programmable data object between the programmable data object and the server processing resource; and

creating a data transfer connection between the programmable data object and the server processing resource.

14. The computer program product according to claim 13, wherein the computing process further comprises creating additional initialization methods to support the items added to the programmable data object.

15. The computer program product according to claim 14, wherein the computing process further comprises:

editing the processing items within the visual design surface module;

*Sub  
all*

updating the processing items edited within visual design surface module;  
 updating the typed dataset within visual design surface module;  
 updating the command adapters within the visual design surface module;  
 identifying any other processing items containing references to data structures  
 and functions edited; and

updating the identified items containing references to data structures and  
 functions edited to make all references consistent with each other.

16. The computer program product according to claim 15, wherein the items  
 comprise properties and processing instruction source code that may be edited.

17. The computer program product according to claim 13, wherein the computing  
 process further comprises:

inserting a database connection module that creates a data transfer connection  
 between the programmable data object and a database when the dragged item is  
 database table within the database.

18. The computer program product according to claim 17, wherein the database  
 connection module comprises:

a data connection object for creating and managing the data transfer connection  
 between the programmable data object and the database;

a managed resource module for provides the data connection object with address  
 and identification information to establish the data transfer connection; and

a persistent data storage for maintaining this address and identification  
 information used by the managed resource module.

a server explorer module for presenting one or more processing resources  
 present on a server to a programmer for use in creating a programming object class;  
 and;

a drag/drop module for enabling a programmer to select a server resource from the server explorer module and place the selected server resource within a data object on the visual design surface module;

a typed dataset generation module for generating typed dataset object associated with the drag and drop of a server processing resource;

a properties edit module for retrieving the properties and source code for the various objects within the visual design surface module for editing.

37

*Handwritten signature*

an explorer interface module to select a server resource from the server explorer module and place it within a data object within the visual design surface module;

a user interface module to perform the visual display and command input operations associated with the drag/drop operation; and

a class generation module to cause the visual design surface module to perform the operations to complete the drag/drop process of a server resource onto the visual design surface module.

21. The system according to claim 19, wherein the drag/drop module further causes the other processing modules in the visual design surface module 402 to perform their operations to complete the drag/drop process of a server resource onto the visual design surface module.

22. The system according to claim 19, wherein the command adapter function generation module comprises:

a GetDS module for generating a GetDataSet function that fills a typed dataset with data obtained from a corresponding database; and

an updateDS module for generating an UpdateDataSet function that updates a database using the data stored within the typed dataset.

23. The system according to claim 19, wherein the command adapter function generation module further accepts an updated command adapter module that has been edited by the properties edit module and generates the updated source code for the functions within the command adapter modules.

24. The system according to claim 19, wherein the typed dataset generation module



comprises:

a Table Schema module for generating the table records from the database schema within the dataset object;

a Relations module for generating the relationship data for the fields within the records within the dataset based upon the corresponding relationship data from the database; and

a Views module for generating the database views data for the records within the dataset based upon the corresponding views data from the database.

25. The system according to claim 19, wherein the typed dataset generation module further accepts an updated typed dataset module that has been edited by the properties edit module and generates the updated source code for the functions within the typed dataset module.

26. The system according to claim 19, wherein the init generation module comprises:

an Init Function module for generating the processing functions and methods within the programmable data object associated with the command adapter modules;

an InitDataSet module for generating the processing functions and methods within a programmable data object module associated with the type dataset class; and

an InitConnection module for generating the processing functions and methods within the programmable data object associated with the data transfer connection between the programmable data object and the database.

27. The system according to claim 19, wherein the init generation module further

[illegible]

updates the identified objects.